Software Reliability Engineering

Werner Dietl <wdietl@uwaterloo.ca>

Table of Contents

This is a draft! This document will be finalized in the first week of the term.

1. Overview

ECE654: Software Reliability Engineering is an **advanced** graduate course. After introducing basic concepts and models for software reliability and security, we will delve into state-of-the-art research and available tools. It is expected that you will read and prepare research papers that will be assigned for the different lectures and that you experiment with the different research prototypes.

The lectures will consist of presentations by the lecturer and the students. Individual projects will complement the lectures and will give first-hand experience with current research.

Topic areas:

- Unit testing, automatic test case generation
- Bug finding tools: Findbugs, PMD, …
- Coding guidelines for reliable and secure software
- Type systems; The Checker Framework
- Design-by-Specification: Spec#, OpenJML, …
- Advanced verification approaches: Dafny, rise4fun, …
- Model checking: Spin, Java Path Finder, NuSVM, …
- Theorem proving: Coq, Isabelle, …

2. Contact

Lectures are held **Thursdays from 11:30 to 14:20 in EIT 3151**[1]. There are no lab or tutorial slots.

My office hours are **by appointment** and will be held in DC 2522. I will also be available immediately after lectures. See contact details[2].

Course material, announcements, and submissions will be handled through Learn[3].

Begin all email subjects with `[ECE654]`.

Try not to leave your questions until the last minute.

2.1. Teaching assistant

TBD.

3. Course content

Prerequisite. Graduate standing in the ECE department at Waterloo.

Advanced programming skills in an object-oriented programming language. Experience in writing medium-sized programs (>20k LOC) and with setting up and using large open-source projects.

Grading. Grades may be curved or adjusted at the instructor’s discretion.

Project: 30%, Midterm: 20%, Final exam: 50%.

The project will consist of a research report (15%), an implementation (10%), and an in-class presentation (5%).

You must pass the final exam, midterm, and project to pass the course.

3.1. Material

There is no textbook that needs to be bought. References for the individual lectures will be made available on Learn.

3.2. Schedule

All lecture material will be available through Learn[4].

This is a tentative schedule that will get adapted during the term.

[1] http://www.adm.uwaterloo.ca/cgi-bin/cgiwrap/infocour/salook.pl?level=grad&sess=1155&subject=ECE&cournum=654
[2] https://ece.uwaterloo.ca/~wdietl/contact.html
[3] https://learn.uwaterloo.ca/d2l/home/197788
[4] https://learn.uwaterloo.ca/d2l/home/158842

Week	Date	Class
1	May 7	Introduction and organization
2	May 14	Background preparation & quiz (instructor absent)
3	May 21	Basic reliability and security concepts and models
4	May 28	Testing, bug finding, and coding guides
5	June 4	Type systems
6	June 11	Midterm (might be moved; decided in 1st week)
7	June 18	Design-by-Specification
8	June 25	Model checking
9	July 2	Theorem proving
10	July 9	Project work (instructor absent)
11	July 16	Project presentations
12	July 23	Outlook and wrap-up

3.3. Project

The project forms an integral part of this course. We will discuss details about the project in the first week of the course and adapt it to the number of students taking the course.

3.4. Expectations

It is expected that students attend lectures and complete the required assignments. Lectures will often include a hands-on activities or presentations by students; participation in these exercises is essential to succeed in the class. Slides will be provided via Learn. Any material discussed in class or in the required readings will be testable unless otherwise noted.

4. Official outline

This is the high-level outline provided by the ECE department[5]; this course will follow the general guideline, but will be adjusted according to your feedback, interests, and experience.

> The course consists of two related parts. The first part deals with the engineering of reliable software. It introduces basic software reliability concepts, describes relevant models and discusses processes for engineering of reliable software, including schemes and patterns for the design of reliable and fault tolerant software. The second part addresses development of secure software. It presents key software security concepts, techniques and models, overviews major software security vulnerabilities and their exploitation, and considers processes for development of secure software.

[5] http://www.ucalendar.uwaterloo.ca/SA/GRAD/1415/GRDcourse-ECE.html#ECE654

5. Policies

Academic Integrity

- In order to maintain a culture of academic integrity, members of the University of Waterloo community are expected to promote honesty, trust, fairness, respect and responsibility. [See the academic integrity[6] site for more information.]

- Plagiarism detection software (Turnitin) will be used to screen assignments in this course. This is being done to verify that use of all materials and sources in assignments is documented. Students will be given an option if they do not want to have their assignment screened by Turnitin. In the first week of the term, details will be provided about arrangements and alternatives for the use of Turnitin in this course.

Grievance

- A student who believes that a decision affecting some aspect of his/her university life has been unfair or unreasonable may have grounds for initiating a grievance. Read Policy 70[7], Student Petitions and Grievances, Section 4.

- When in doubt please be certain to contact the department's administrative assistant who will provide further assistance.

Discipline

- A student is expected to know what constitutes academic integrity[8] to avoid committing an academic offence, and to take responsibility for his/her actions.

- A student who is unsure whether an action constitutes an offence, or who needs help in learning how to avoid offences (e.g., plagiarism, cheating) or about "rules" for group work/collaboration should seek guidance from the course instructor, academic advisor, or the undergraduate Associate Dean.

- For information on categories of offences and types of penalties, students should refer to Policy 71[9], Student Discipline.

- For typical penalties check Guidelines for the Assessment of Penalties[10].

Appeals

- A decision made or penalty imposed under Policy 70 (Student Petitions and Grievances) (other than a petition) or Policy 71 (Student Discipline) may be appealed if there is a ground.

[6] http://www.uwaterloo.ca/academicintegrity/
[7] http://www.adm.uwaterloo.ca/infosec/Policies/policy70.htm
[8] http://www.uwaterloo.ca/academicintegrity/
[9] http://www.adm.uwaterloo.ca/infosec/Policies/policy71.htm
[10] http://www.adm.uwaterloo.ca/infosec/guidelines/penaltyguidelines.htm

- A student who believes he/she has a ground for an appeal should refer to Policy 72[11], Student Appeals.

Note for Students with Disabilities

- AccessAbility Services[12], located in Needles Hall, Room 1132, collaborates with all academic departments to arrange appropriate accommodations for students with disabilities without compromising the academic integrity of the curriculum. If you require academic accommodations to lessen the impact of your disability, please register with the AccessAbility Services at the beginning of each academic term.

6. Acknowledgments

Thanks to Mohammad Ahmad Munawar and Rudolph E. Seviora for sharing their course material for previous iterations of the course.

[11] http://www.adm.uwaterloo.ca/infosec/Policies/policy72.htm

[12] https://uwaterloo.ca/disability-services/